Release Notes

CodeWarrior™ Development Studio for Microcontrollers V10.1

TABLE OF CONTENTS

1. Product Overview 3

1.1 IDE 3

1.2 Wizards 3

1.3 Build Tools 3

1.3.1 Kinetis 3

1.3.2 RS08/HCS08 4

1.3.3 ColdFire 4

1.3.4 MPC56xx 4

1.3.5 General 4

1.4 Debugger 4

1.4.1 Kinetis 4

1.4.2 MPC56xx 5

1.4.3 ColdFire 5

1.4.4 General 5

1.5 Flash Programming 5

1.5.1 MPC56xx 5

1.5.2 Flash Programmer 5

1.5.3 General 5

1.6 Target Connections 5

1.6.1 Kinetis target connections 5

1.6.2 MPC56xx target connections 5

1.6.3 General 5

1.7 Software Analysis 6
1.8 Processor Expert 6
1.9 Documentation 6
1.9.1 Getting Started 6
1.9.2 Users Guides 6
1.9.3 Application Notes 7
1.9.4 Supporting Information 7
1.9.5 Cheat Sheets (updated) 7
1.5.6 Videos (available at www.freescale.com/cwmcu10) 8
2 System Requirements 8
2.1 Recommended Configuration 8
2.2 Operational Minimum Configuration 8
2.3 Host Operating System Support 8
3 Product WEB page 8
4 Installation and Licensing 8
5 Technical Support 9
Appendix A: Known issues and Workarounds 10
Appendix B: CodeWarrior Eclipse usage on a Windows Vista or Windows 7 system 11
Appendix C: CodeWarrior Eclipse usage on a Linux system 11
Appendix D: Installing P&E Microcomputer Systems' USB drivers on a Linux system 11

1. Product Overview

Freescale's CodeWarrior for Microcontrollers v10.1 integrates the development tools for the RS08, HCS08, ColdFire®, ColdFire+, Kinetis and MPC56xx architectures into a single product based on the Eclipse open development platform. Eclipse offers an excellent framework for building software development environments and is becoming a standard framework used by many embedded software vendors. The new product features include the following:

1.1 IDE

- Added ability to disassemble ELF object files or absolute executable files in a project quickly
- Modified/new files will automatically be saved before a manual build. This functionality can be changed in the Windows>Preferences>General>Workspace menu.
- Added Terminal view which can be used as a console to a COM port
- Updated build output console to navigate to error source code by double-clicking on an error message.
- Make files may be exported for use outside of Eclipse.
- "About Dialog" updated to display CodeWarrior Development Studio version
- Welcome screen includes scroll bars to allow content to be displayed on small screens
- Welcome screen provides access to example projects
- Help menu includes menu items for the following:
    - User Assistance Map
    - About CodeWarrior Development Studio
- Added support for Ubuntu 8.x host-operating system
- Added support for Linux 64-bit host-operating systems

1.2 Wizards

- New Project Wizard adds support for the following
    - New connections
    - Remote System Explorer (RSE)
    - Kinetis derivatives
    - MPC56xx derivatives
    - DPM/LSM feature on MPC56xx derivatives

- Change Wizard adds support for the following:
    - New connections
    - Remote System Explorer (RSE)
    - Kinetis derivatives

- ELF files from existing MPC56xx projects can be imported into CW MCU v10.1.with the ELF Importer.

1.3 Build Tools

1.3.1 Kinetis

- Full-featured ARM Cortex-M4 ANSI C/C++ compiler, linker, assembler, and standard C/C++ and runtime libraries.
    - Function-level and statement-level inline assembly support for the Cortex-M4 instruction set in the compiler
    - A standalone assembler for the Cortex-M4 instruction set
    - Intrinsic library support for the Cortex-M4 DSP instructions
    - Full ARM EABI 2 conformance
    - ELF/DWARF output formats, including DWARF3

- Comprehensive command-line interface and full integration with the Eclipse IDE for all tools
- Position Independent Code(PIC)/Position Independent Data (PID) code generation support
- Optimizations for speed and size
- Advanced optimizations such as inlining and loop unrolling
- Object file dumping and disassembly utilities

1.3.2 RS08/HCS08

- Linker computes maximum stack consumption for applications.
- Linker can create an empty section in prm file.
- HCS08 compiler optimizes successive writes of the same 1-byte constant into zero-page memory.
- Environment variables can be expanded in the command line.
- Support for additional command-line arguments in input files.
- Compilation speed with make files has been improved by ~20%
- New inline assembly directive (__RETADDR) added to improve debugger usability. For more information see the RS08/HCS08 compiler release notes.

1.3.3 ColdFire

- Added support for raw-binary file generation
- Improved debug information generation for optimized code
- Improved optimized code generation
- "if (ptr == NULL)" constructions branch/condition are optimized in the generated code
- Calling convention option provided to allow decision to be made during preprocessing
- Using new flavor of strtod() in the compiler FrontEnd, allowing a more broad range of accepted / parsed floating point values
- Added ColdFire Embedded Warrior Library (EWL) including eclipse projects for rebuilding the ColdFire EWL libraries
- Compilation speed with make files has been improved by ~20%

1.3.4 MPC56xx

- ANSI C/C++ compiler, linker, assembler, disassembler and C/C++ and runtime library supported via the Eclipse IDE and command-line
- The .rela.* sections are no longer generated by default. These sections are generated for the profiler CATS. To generate these sections, please use “#pragma cats on”
- "-pragma fp_contract off" can be used to disable the SPE additional fused multiply-add instructions codegen when -spe2_vector is selected. By turning off this optimization, the floating point accuracy will be maintained.
- "-pragma fp_contract_aggressive on" can be used to further optimize multiply-add opportunities. Precision could be lost due to rounding issues.

1.3.5 General

- Added support for different extensions with same base file name, e.g. main.asm and main.c.

1.4 Debugger

1.4.1 Kinetis

- Standard run-control (run, stop, step in, step out, go and setting breakpoints)
- Ability to display all available processor general purpose, special purpose and memory mapped registers with bit-level information
- Ability to decode and display ARM Cortex-M4 THUMB2 instructions
- Debug support for TWR-K40X256 and TWR-K60N512 boards
- Kernel-aware debug support for MQX (plug-in distributed with MQX)

1.4.2 MPC56xx

- Standard run-control (run, stop, step in, step out, go and setting breakpoints)
- Ability to display all available core and memory mapped registers, Nexus registers, MMU registers (TLBs)
- Support for multi-core debugging including Lock Step Mode (LSM), Dual Processor Mode (DPM) and multi-core operations
- Kernel-aware debug support for AUTOSAR/OSEK

1.4.3 ColdFire

- Linux uboot, kernel and application debug for ColdFire TWR-MCF5441X board.

1.4.4 General

- Disassembler allows unlimited scrolling in either direction and provides stepping history
- Source Lookup for multiple launches is available
- Automatically migrates launch configurations created with previous versions of CodeWarrior
- Same ELF file can be used for different Download Launch Configurations
- Added support for TCL based initialization scripts
- Kernel Awareness SDK: Developer Guide is available

1.5 Flash Programming

1.5.1 MPC56xx

- All basic flash operations are supported for internal flash modules: code, data, shadow flash and test areas
- Switching between Locked Step Mode (LSM) and Dual Processor Mode (DPM) with predefined target tasks
- Autoflash support (direct download to flash or via target task framework)

1.5.2 Flash Programmer

- Ability to program one section of a block
- Ability to send sector index to the flash programming algorithm
- Ability to secure/unsecure device

1.5.3 General

- Flash programming support for external and on-chip flash memory (NAND, NOR, SPI)
- Protect and secure sections of flash memory
- Flash programming SDK to allow creation of custom flash algorithms

1.6 Target Connections

1.6.1 Kinetis target connections

- P&E USB Multilink
- P&E Cyclone MAX
- Segger J-Link
- Open Source JTAG

1.6.2 MPC56xx target connections

- P&E Qorivva Multilink
- P&E Cyclone MAX
- Open Source JTAG

1.6.3 General

- Target management updated to use Remote System Explorer (RSE)

1.7 Software Analysis

- Added ARM trace support including:
  - ETM, ITM, DWT, ETB trace configuration;
  - Trace and profile results manager;
  - Trace viewer;
  - Critical Code (flat profiler) viewer;
  - Hardware tracepoints
- Added ColdFire V2-V4 simple profiler

1.8 Processor Expert

- Added code generation log to provide information about changes made during code generation.
- Kinetis:
  - Added support for Kinetis K10, K20, K30, K40, K60 derivative families in 144MAPBGA package.
  - Added new class of embedded components – Logical Device Drivers (LDD). These components offer a Hardware Abstraction Layer (HAL) for bare-metal applications as well as RTOS applications.
  - Introduced shared mode for LDD components to allow reuse of the same code module by several components.
  - Added MQX RTOS adapter to allow LDD components to be used in a MQX board support package (BSP).
  - Initialization components support, Device initialization support.
- (R)S08/V1ColdFire:
  - Added support for new S08 derivatives: GW64, GW32, ER48
  - Added support for new Peripheral Initialization components: Init_IRTC, Init_PCNT, Init_CRC, Init_RFM, Init_SAW
- V2/V3/V4 ColdFire
  - Added support for MCF5441x derivative family

1.9 Documentation

CodeWarrior Development Studio for Microcontrollers V10.1 contains the following new or updated documentation since Microcontrollers 10.0.

1.9.1 Getting Started

- Microcontrollers V10.x Getting Started Guide **(updated)**
- Quick Start for Microcontrollers V10.x **(updated)**
- Quick Start for CodeWarrior Project Importer **(updated)**
- Eclipse Quick Reference Card **(updated)**
- HCS08 Profiling and Analysis for Microcontrollers V10.x Quick Start **(new)**
- ColdFire V1 Profiling and Analysis for Microcontrollers V10.x Quick Start **(new)**
- Quick Start for Ethernet TAP Probe **(new)**

1.9.2 Users Guides

- Freescale Eclipse Extensions Guide **(updated)**
- Microcontrollers V10.x Targeting Manual **(updated)**
- HCS08 Build Tools Reference Manual for Microcontrollers 10.x **(updated)**
- RS08 Build Tools Reference Manual for Microcontrollers 10.x **(updated)**
- ColdFire Build Tools Reference Manual for Microcontrollers 10.x **(updated)**
- Kinetis Build Tools Reference Manual for Microcontrollers 10.x **(new)**
- Power Architecture Processors Build Tools Reference Manual for Microcontrollers V10.x **(new)**

- MISRA-C: 2400 Compliance Exceptions for HCS08, RS08, ColdFire, Kinetis and Power Architectures Processors Library for Microcontrollers V10.x **(new)**
- EWL C Reference Manual
- EWL C++ Reference Manual **(updated)**
- HCS08/RS08 Assembler Reference Manual for Microcontrollers V10.x **(updated)**
- ColdFire Assembler Reference Manual for Microcontrollers V10.x **(updated)**
- Kinetis Assembler Reference Manual for Microcontrollers V10.x **(new)**
- HC(S)08/RS08 Build Tools Utilities Manual for Microcontrollers V10.x **(updated)**
- Profiling and Analysis User Guide for Microcontrollers V10.x **(updated)**
- USB TAP Probe Users Guide
- Ethernet TAP Probe Users Guide
- Open Source BDM-JM60 Users Guide
- Processor Expert User Manual **(updated)**
- Device Initialization User Manual **(updated)**
- Signal Processing Engine Auxiliary Processing Unit Programming Interface Manual **(new)**

1.9.3 Application Notes

- AN3859 – Adding Device(s) to the CodeWarrior Flash Programmer for Microcontrollers V10.x
- AN3967 – How to Write Flash Programming Applets
- AN4095 – CodeWarrior Build Tools Options for Optimal Performance on the Power Architecture e200 Core **(new)**
- AN4104 – Converting ColdFire Projects to CodeWarrior Development Studio for Microcontrollers 10.x **(new)**
- AN4188 – RS08 Upper Memory Access **(new)**

1.9.4 Supporting Information

- Microcontrollers V10.x FAQ Guide **(updated)**

1.9.5 Cheat Sheets **(updated)**

- Microcontroller V10.x Specific
  - Building Library (HXS08)
  - Creating, Building and Debugging a Project
  - Creating, Debugging, Collecting and Viewing Data for the ColdFire V1 Target
  - Creating, Debugging, Collecting and Viewing Data for the ColdFire V1 Target
  - Creating, Debugging, Collecting and Viewing Data for the HCS08 Target
  - Creating New Project from Example Project
  - Changing P&E Connections Setting
  - Configuring Perspectives
  - Debugging Projects in ROM
  - Debugging Project Using Command Line
  - Examples: Porting Classic IDE Projects to Eclipse
  - Importing and Debugging Externally Built Executable File
  - Using Memory View
  - Using Microcontrollers Change Wizard
  - Using Registers View
  - Working with Build Configurations
- Core (Common to other CodeWarrior products)
  - Importing Archive Files
  - Importing Break Points
  - Importing Classic CodeWarrior Projects
  - Importing Existing Projects into Workspace
  - Importing Preferences
  - Importing Resources from Local File Systems

- Importing Team Project Set
- Making C/C++ the IDE's Default Perspective
- Using the Flash Programmer

1.5.6 Videos (available at www.freescale.com/cwmcu10)

- CodeWarrior Overview
- How to Debug a Project
- Importing a Project
- Critical Code
- Creating a Project from and Example Project
- Changing Build Configurations
- Adding Registers View
- Adding Memory Monitors How to Debug a Target Without an Application

2 System Requirements

2.1 Recommended Configuration

- 2.6 GHz Pentium® compatible processor or better
- Microsoft Windows XP/Vista
- 2 GB RAM
- 2 GB hard disk space, 400 MB on Windows system disk
- CD-ROM drive for installation
- USB port for communications with target hardware
- Ethernet port for communications with target hardware (optional)

2.2 Operational Minimum Configuration

- 1.8 GHz Pentium® compatible processor or better
- Microsoft Windows XP
- 1 GB RAM
- 1 GB hard disk space, 400 MB on Windows system disk
- CD-ROM drive for installation
- USB port for communications with target hardware

2.3 Host Operating System Support

- Microsoft® Windows XP 32-bit and 64-bit (Professional Edition)
- Microsoft Windows Vista® 32-bit and 64-bit (Home Premium Edition and Business Edition)
- Microsoft Windows 7 32-bit and 64-bit (Home Premium Edition and Professional Edition)
- Red Hat Enterprise Edition 5.2 (32-bit and 64-bit)
- Ubuntu 8.x (32-bit and 64-bit)

3 Product WEB page

CodeWarrior Development Studio for Microcontrollers v10.1 is available for download at http://www.freescale.com/cwmcu10. This web page also includes product information, task based videos and all related documentation.

4 Installation and Licensing

To install CodeWarrior Development Studio for Microcontrollers v10.1, double-click the installation package and a wizard will guide you through the installation process.

If you are installing the Special Edition, the Special Edition license is automatically installed with your product and you do not need to register it. This license allows you to develop projects with unlimited assembly code, up to 32KB of C code for HCS08/RS08 derivatives, up to 64KB of C code for V1 ColdFire derivatives, up to 128KB of C code for V2-V4 ColdFire and Kinetis derivatives and up to 512KB of C code for MPC56xx derivatives.

If you are installing the Evaluation Edition, the Evaluation license is automatically installed with your product and you do not need to register it. This license allows you to develop projects as Professional Edition within the 30-day evaluation period. After 30 days, the license works as Special Edition license (free permanent, but feature limited) which supports unlimited assembly code, up to 32KB of C code for HCS08/RS08 derivatives, up to 64KB of C code for V1 ColdFire derivatives and up to 128KB of C code for V2-V4 ColdFire and Kinetis derivatives and up to 512KB of C code for MPC56xx derivatives.

If you are installing the Basic, Standard, or Professional Editions, the Evaluation license will be installed with your product. You will need to register your product to receive the appropriate license file.

5 Technical Support

All CodeWarrior issues are tracked through Freescale's normal Service Request Process. To report feature requests (enhancements) or defects for CodeWarrior Development Studio for Microcontrollers v10.1, please submit a Service Request.

1. Go to http://www.freescale.com/support
2. Log in.
3. On the resulting MyFreescale page, click Enter a Service Request
4. Choose category Technical Request
5. Choose topic CodeWarrior
6. Click Continue.
7. Provide the required information. An attachment up to 10 MB may be attached to the SR. You may also specify email addresses of people you would like to keep notified on the progress of the SR. Separate multiple email addresses with commas.

    - **Type:** pick from Query (Question), Change Request (Enhancement) or Problem Report (Bug)
    - **Target:** specify the hardware microcontroller/microprocessor family involved
    - **Product:** CW for Microcontrollers
    - **Major:** 10
    - **Minor:** 1
    - **Priority:** choose from Normal, High, or Highest
    - **Subject:** be short and descriptive
    - **Description:** details about whatever it is you're filing the SR about

    Please note: The Product field must be set to CW for Microcontrollers. This will allow Freescale to find SRs related to this project very easily, report on them, and gather statistics on how the project is doing.

8. When finished, click Submit.

After Submit is selected, a confirmation page will be displayed with the SR number. You will also receive a confirming email sent to the address specified in your Freescale account.

Appendix A: Known issues and Workarounds

Issue ID	Description
IDE	
MTWX45661	**Issue:** When an Eclipse workspace with at least one project is started, the "Remote System Changed" dialog will be displayed for the RSE system. **Workaround:** None. Click OK to close the dialog.
Debugger	
MTWX44205	**Issue:** Memory configuration files and download task files (debugger shell task) with Japanese and Chinese characters (UTF16 encoded characters) in the directory path will cause a "Program not recognized" error **Workaround:** Ensure memory configuration files and download task files are not in directory paths which include UTF16 encoded characters.
MTWX45588	**Issue:** Can't attach to a running MPC56xx target. **Workaround:** This feature is not supported in this release.
MTWX45625	**Issue:** Cannot program restricted memory areas on S08 paged memory devices with Flash Programmer Target Task. **Workaround:** None. Flash Programmer Target Task can only program entire flash.
Build Tools	
MTWX45507	**Issue:** The ARM build tools might generate the wrong debug information for constructor/destructor functions in C++ projects. **Workaround:** When a breakpoint cannot be set in a source or header file (even though there are no optimizations or inline functions) set an address breakpoint in the disassembly view instead.
MTWX45550	**Issue:** A build error is reported in the Console View for any project with a long directory path. **Workaround:** Ensure the filename and directory path do not exceed 255 characters. In the Windows API the maximum length for a path is MAX_PATH, which is defined as 260 characters. A local path is structured in the following order: drive letter, colon, backslash, name components separated by backslashes, and a terminating null character. For more information see http://msdn.microsoft.com/en-us/library/aa365247(v=vs.85).aspx#maxpath
Project Importer	
MTWX43487 MTWX44755	**Issue:** Project Importer does not convert V2-V4 ColdFire linker files to new object file naming convention. **Workaround:** When importing V2-V4 ColdFire projects (CW MCU v10.0, CW CF v7.x), the content of the project's LCF file must be modified to replace any reference to object or source files with the correct reference to their corresponding object files (considering that the extension of the source files should be included in the name of the generated object file): ○ references to "a.c" should be replaced with "a_c.obj" ○ references to "b.asm" should be replaced with "b_asm.obj" ○ references to "d.cpp" should be replaced with "d_cpp.obj" ○ references to "e.obj" should be replaced with "e_ext.obj" (if the corresponding source file is "e.ext")

MTWX45630	**Issue:** CW MCU v6.3 Smart Linker message settings ("-Wmsg" flags) do not reach the linker settings in the Eclipse project build configurations. **Workaround:** Manually add the message disabling options to the linker command line.
Wizards	
MTWX45639	**Issue:** Cannot build ColdFire V2-V4 Static Library/ Shared Library on a Linux hosted machine if there is a space in the directory path. **Workaround:** Ensure all directory paths for include files do NOT contain spaces.

Appendix B: CodeWarrior Eclipse usage on a Windows Vista or Windows 7 system

1. The CodeWarrior installer should be run using the 'Run as administrator' option. CodeWarrior service packs are installed with the Eclipse Updater. The updater should also be run with this option. To start the Eclipse Updater select 'Window > Install new software' in the menu.

2. Eclipse needs read/write access to the installation folder. Make sure the eclipse installation folder has the appropriate permissions for all users.

3. Make sure your project workspace has read and write permissions.

Appendix C: CodeWarrior Eclipse usage on a Linux system

1. The CodeWarrior installer should be run from a root account. CodeWarrior service packs are installed with the Eclipse Updater. The updater should also be run from a root account. To start the Eclipse Updater select 'Window > Install new software' in the menu.

2. Eclipse needs read/write access to the installation folder. Make sure the eclipse installation folder has the appropriate permissions for all users.

3. Make sure your project workspace has read and write permissions.

Appendix D: Installing P&E Microcomputer Systems' USB drivers on a Linux system

To properly install P&E USB drivers to be on a Linux system, the user should verify the following requirements are satisfied before installation (either manually or via CodeWarrior):

A. root access must be used during install.
B. kernel source must be included in the distribution (same version used to build the kernel).
C. GCC must be installed on the machine (same version as used to build the kernel).

See Section 2 for directions how to manually install the kernel source and GCC.

1. Determining whether the drivers have been installed properly

If driver installation fails, a warning message should be displayed during installation. The user can also look in the /usr/lib directory for the file libwdusb1002.so. If this file exists, driver installation was successful.

If the driver installation was unsuccessful, the drivers must be manually installed.

2. Manually installing GCC and kernel source on redhat 5

The same version of GCC, which was used to build the kernel, must be installed in the distribution. The same kernel source files, which were used to build the kernel, must also be installed (or at least the headers). The following steps demonstrate how to install the necessary resources on Red Hat 5 Enterprise Edition (32-bit).*

A. Open a console (terminal) window and use the 'su' command to enter root mode

B. Use the 'cat /proc/version' command to display the kernel version number and the GCC version number used to build the kernel.

C. Use the 'yum list gcc' command to display the version number of GCC in the yum repository. If it matches the version used to build the kernel, install it with the 'yum install gcc' command. Otherwise, download and install the appropriate GCC version from an on-line repository.

D. Use the 'yum list kernel-devel' command display the version number of the kernel source files in the yum repository. If it matches the kernel version number, install it with the 'yum install kernel-version' command. Otherwise, download and install the source from an on-line repository.

E. Change directory to the distribution source directory with the 'cd /usr/src' command.

F. Create a symbolic link which points to the source for this kernel. This is done by first using the 'ls kernels' command while in the /usr/src directory. This will display the version of source installed, such as 2.6.18-128.2.1.el5-i686. create the symbolic link with the command 'ln -s kernels/2.6.18-128.2.1.el5-i686 linux' where the version number of the kernel source would be replaced with the appropriate version number listed by the previous command.

For information about the driver installation requirements visit Jungo's website at:
http://www.jungo.com/st/support/installation_instructions.html

3. Manually installing the Jungo USB driver on redhat 5

After making sure that GCC and kernel source have been installed, the Jungo driver should be rebuilt/installed. The following are directions for doing this on redhat 5.

A. Open a console (terminal) window and use the 'su' command to enter root mode

B. If codewarrior has not been installed, install it. This should install the Jungo USB driver (see section 2 to determine if it has been properly installed).

C. To re-install the Jungo USB driver, change the console directory to the [CodewarriorInstallDir]/Drivers/pemicro. Run the driver build script by using the 'sh setup.sh' command. This will indicate success or failure.

* There are many ways to install the kernel source and GCC, and the methodology is different based on the distribution used.